Clustering

COMS 4771
1. Clustering
Unsupervised classification / clustering

Unsupervised classification

- **Input**: $x_1, \ldots, x_n \in \mathbb{R}^d$, target cardinality $k \in \mathbb{N}$.
- **Output**: function $f : \mathbb{R}^d \rightarrow \{1, \ldots, k\} =: [k]$.
- **Typical semantics**: hidden subpopulation structure.

Clustering

- **Input**: $x_1, \ldots, x_n \in \mathbb{R}^d$, target cardinality $k \in \mathbb{N}$.
- **Output**: partitioning of $x_1, \ldots, x_n$ into $k$ groups.

- Often done via unsupervised classification; ⇒ "clustering" often synonymous with "unsupervised classification".
- Sometimes also have a "representative" $c_j \in \mathbb{R}^d$ for each $j \in [k]$ (e.g., average of the $x_i$ in $j$th group) → quantization.
Unsupervised classification / clustering

Unsupervised classification

- **Input**: $x_1, \ldots, x_n \in \mathbb{R}^d$, target cardinality $k \in \mathbb{N}$.
- **Output**: function $f : \mathbb{R}^d \to \{1, \ldots, k\} =: [k]$.
- **Typical semantics**: hidden subpopulation structure.

Clustering

- **Input**: $x_1, \ldots, x_n \in \mathbb{R}^d$, target cardinality $k \in \mathbb{N}$.
- **Output**: partitioning of $x_1, \ldots, x_n$ into $k$ groups.
- Often done via unsupervised classification;
  \[ \Rightarrow \text{“clustering” often synonymous with “unsupervised classification”}. \]
- Sometimes also have a “representative” $c_j \in \mathbb{R}^d$ for each $j \in [k]$ (e.g., average of the $x_i$ in $j$th group) \[\rightarrow\text{quantization}.\]
Unsupervised classification / clustering

Input: \( x_1, \ldots, x_n \in \mathbb{R}^d \), target cardinality \( k \in \mathbb{N} \).

Output: partitioning of \( x_1, \ldots, x_n \) into \( k \) groups.

Often done via unsupervised classification; \( \Rightarrow \) “clustering” often synonymous with “unsupervised classification”.

Sometimes also have a “representative” \( c_j \in \mathbb{R}^d \) for each \( j \in [k] \) (e.g., average of the \( x_i \) in \( j \)th group) \( \rightarrow \) quantization.
“One-hot” / “dummy variable” encoding of \( f(x) \)

\[
\phi(x) = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}
\]

(Often used together with other features.)
Uses of clustering: feature representations

Histogram representation

- Cut up each $x_i \in \mathbb{R}^d$ into different parts $x_{i,1}, \ldots, x_{i,m} \in \mathbb{R}^p$ (e.g., small patches of an image).

- Cluster all the parts $x_{i,j}$: get $k$ representatives $c_1, \ldots, c_k \in \mathbb{R}^p$.

- Represent $x_i$ by a histogram over $\{1, \ldots, k\}$ based on assignments of $x_i$’s parts to representatives.
Uses of clustering: compression

**Quantization** Replace each $x_i$ with its representative

$$x_i \mapsto c_f(x_i).$$

**Example:** quantization at image patch level.
2. $k$-means
\(k\)-means optimization problem

- **Input**: \(x_1, \ldots, x_n \in \mathbb{R}^d\), target cardinality \(k \in \mathbb{N}\).
- **Output**: \(k\) “means” \(c_1, \ldots, c_k \in \mathbb{R}^d\).
- **Objective**: choose \(c_1, \ldots, c_k \in \mathbb{R}^d\) to minimize

\[
\sum_{i=1}^{n} \min_{j \in [k]} \|x_i - c_j\|_2^2.
\]
**$k$-means optimization problem**

- **Input:** $x_1, \ldots, x_n \in \mathbb{R}^d$, target cardinality $k \in \mathbb{N}$.
- **Output:** $k$ “means” $c_1, \ldots, c_k \in \mathbb{R}^d$.
- **Objective:** choose $c_1, \ldots, c_k \in \mathbb{R}^d$ to minimize

$$\sum_{i=1}^{n} \min_{j \in [k]} \|x_i - c_j\|_2^2.$$ 

**Natural assignment function**

$$f(x) := \arg \min_{j \in [k]} \|x - c_j\|_2^2.$$
**k**-means optimization problem

- **Input:** \( x_1, \ldots, x_n \in \mathbb{R}^d \), target cardinality \( k \in \mathbb{N} \).
- **Output:** \( k \) “means” \( c_1, \ldots, c_k \in \mathbb{R}^d \).
- **Objective:** choose \( c_1, \ldots, c_k \in \mathbb{R}^d \) to minimize

\[
\sum_{i=1}^{n} \min_{j \in [k]} \| x_i - c_j \|_2^2.
\]

**Natural assignment function**

\[
f(x) := \arg \min_{j \in [k]} \| x - c_j \|_2^2.
\]

**NP-hard, even if \( k = 2 \) or \( d = 2 \).**
$k$-means clustering for $k = 1$

**Problem:** Pick $c \in \mathbb{R}^d$ to minimize

$$\sum_{i=1}^{n} \|x_i - c\|_2^2.$$
The easy cases

> **k-means clustering for** $k = 1$

**Problem**: Pick $c \in \mathbb{R}^d$ to minimize

$$\sum_{i=1}^{n} \|x_i - c\|_2^2.$$

**Solution**: Using “bias/variance decomposition”, best choice is $c = \frac{1}{n} \sum_{i=1}^{n} x_i$. 
The easy cases

- **k-means clustering for** $k = 1$
  
  **Problem:** Pick $c \in \mathbb{R}^d$ to minimize
  \[
  \sum_{i=1}^{n} \|x_i - c\|_2^2.
  \]

  **Solution:** Using “bias/variance decomposition”, best choice is
  \[
  c = \frac{1}{n} \sum_{i=1}^{n} x_i.
  \]

- **k-means clustering for** $d = 1$
  
  Dynamic programming in time $O(n^2k)$. 

Lloyd's algorithm is a popular heuristic for general $k$ and $d$.
(There are many other algorithms for $k$-means clustering.)
The easy cases

- **$k$-means clustering for $k = 1$**

  **Problem**: Pick $c \in \mathbb{R}^d$ to minimize

  \[
  \sum_{i=1}^{n} \| x_i - c \|_2^2.
  \]

  **Solution**: Using “bias/variance decomposition”, best choice is $c = \frac{1}{n} \sum_{i=1}^{n} x_i$.

- **$k$-means clustering for $d = 1$**

  Dynamic programming in time $O(n^2k)$.

  Lloyd’s algorithm is a popular heuristic for general $k$ and $d$.

  (There are many other algorithms for $k$-means clustering.)
Lloyd’s algorithm

- Start with some initial “means” \( c_1, \ldots, c_k \).

- Repeat:
  - Partition \( x_1, \ldots, x_n \) into \( k \) clusters \( C_1, \ldots, C_k \), based on distance to current “means”:
    \[
    x_i \mapsto \arg \min_{j \in [k]} \| x_i - c_j \|_2^2, \quad i \in [n],
    \]
    breaking ties according to any fixed rule.

- Update “means”:
    \[
    c_j := \text{mean}(C_j), \quad j \in [k].
    \]
Sample run of Lloyd’s algorithm

Arbitrary initialization of $c_1$ and $c_2$. 
Sample run of Lloyd’s algorithm

Iteration 1
Optimize “assignments”.
Sample run of Lloyd’s algorithm

Iteration 1
Optimize “means” \( c_j \).
Sample run of Lloyd’s algorithm

Iteration 2
Optimize “assignments”.
Sample run of Lloyd’s algorithm

Iteration 2
Optimize “means” $c_j$. 
Sample run of Lloyd’s algorithm

Iteration 3
Optimize “assignments”.

-2 0 2
-2
0
2
Sample run of Lloyd’s algorithm

Iteration 3
Optimize “means” $c_j$. 
Sample run of Lloyd’s algorithm

Iteration 4
Optimize “assignments”.

$h$

−2 0 2
−2
0
2

(h)
Sample run of Lloyd’s algorithm

Iteration 4
Optimize “means” $c_j$. 
**Basic idea:** Choose initial centers to have good coverage of the data points.
Initializing Lloyd’s algorithm

**Basic idea:** Choose initial centers to have good coverage of the data points.

**Farthest-first traversal**
For $j = 1, \ldots, k$:

- Pick $c_j \in \mathbb{R}^d$ from among $x_1, \ldots, x_n$ farthest from previously chosen $c_1, \ldots, c_{j-1}$.
  
  ($c_1$ chosen arbitrarily.)
Basic idea: Choose initial centers to have good coverage of the data points.

Farthest-first traversal
For \( j = 1, \ldots, k \):

- Pick \( c_j \in \mathbb{R}^d \) from among \( x_1, \ldots, x_n \) farthest from previously chosen \( c_1, \ldots, c_{j-1} \).

(\( c_1 \) chosen arbitrarily.)

But this can be thrown off by outliers...
**Initializing Lloyd’s algorithm**

**Basic idea:** Choose initial centers to have good coverage of the data points.

**Farthest-first traversal**
For $j = 1, \ldots, k$:

- Pick $c_j \in \mathbb{R}^d$ from among $x_1, \ldots, x_n$ farthest from previously chosen $c_1, \ldots, c_{j-1}$.

  ($c_1$ chosen arbitrarily.)

But this can be thrown off by outliers...

**$D^2$ sampling (a.k.a. “$k$-means++”)**
For $j = 1, \ldots, k$:

- Randomly pick $c_j \in \mathbb{R}^d$ from among $x_1, \ldots, x_n$ according to distribution

  $$P(x_i) \propto \min_{\ell=1,\ldots,j-1} \|x_i - c_\ell\|_2^2.$$  

  (Uniform distribution when $j = 1$.)
Choosing $k$

- Usually by hold-out validation / cross-validation on auxiliary task (e.g., supervised learning task).
Choosing $k$

- Usually by hold-out validation / cross-validation on auxiliary task (e.g., supervised learning task).

- Heuristic: Find large gap between $(k - 1)$-means cost and $k$-means cost.
3. Hierarchical clustering
Clustering at multiple scales

$k = 2$ or $k = 3$?

Hierarchical clustering: encode clusterings for all values of $k$ in a tree.

Caveat: not always possible.
Clustering at multiple scales

$k = 2$ or $k = 3$?

*Hierarchical clustering*: encode clusterings for all values of $k$ in a tree.
Clustering at multiple scales

$k = 2$ or $k = 3$?

*Hierarchical clustering*: encode clusterings for all values of $k$ in a tree.

*Caveat*: not always possible.
Clustering at multiple scales

$k = 2$ or $k = 3$?

*Hierarchical clustering:* encode clusterings for all values of $k$ in a tree.

*Caveat:* not always possible.
Clustering at multiple scales

$k = 2$ or $k = 3$?

*Hierarchical clustering:* encode clusterings for all values of $k$ in a tree.

*Caveat:* not always possible.
Example: phylogenetic tree
Agglomerative clustering

- Start with every point $x_i$ in its own cluster.
- Repeatedly merge “closest” pair of clusters, until only one cluster remains.
Agglomerative clustering

- Start with every point \( x_i \) in its own cluster.
- Repeatedly merge “closest” pair of clusters, until only one cluster remains.

**Single-linkage**

\[
\text{dist}(C, C') := \min_{x \in C, x' \in C'} \| x - x' \|_2.
\]

**Complete-linkage**

\[
\text{dist}(C, C') := \max_{x \in C, x' \in C'} \| x - x' \|_2.
\]
Agglomerative clustering

- Start with every point $x_i$ in its own cluster.
- Repeatedly merge “closest” pair of clusters, until only one cluster remains.

**Single-linkage**

$$\text{dist}(C, C') := \min_{x \in C, x' \in C'} \|x - x'\|_2.$$  

**Complete-linkage**

$$\text{dist}(C, C') := \max_{x \in C, x' \in C'} \|x - x'\|_2.$$  

**Average-linkage**

Many variants. E.g., *Ward’s average linkage*

$$\text{dist}(C, C') := \frac{|C| \cdot |C'|}{|C| + |C'|} \|\text{mean}(C) - \text{mean}(C')\|_2^2.$$
Key takeaways

- **Uses of clustering:**
  - Unsupervised classification ("hidden subpopulations").
  - Quantization
  - ...
  - $k$-means clustering: popular objective for clustering and quantization.
  - Lloyd’s algorithm: alternating optimization, needs good initialization.
  - Hierarchical clustering: clustering at multiple levels of granularity.